

# Journal de développement

## Choix faits durant le développement

Afin de faciliter le développement de notre moteur de jeu, nous avons décidé d'utiliser l'IDE Clion qui appartient à la suite de logiciel JetBrains.

Cet IDE dispose de plusieurs fonctionnalités utiles pour le développement comme la génération de code ou l'auto-complétion. De plus, l'ensemble des membres de notre groupe a eu l'habitude d'utiliser Clion pour les projets de C++ effectués au cours de notre formation.

Nous avons ensuite choisi GoogleTest comme framework de test pour notre moteur de jeu. L'utilisation d'un framework de test nous permet de pouvoir faire des tests de manière rapide et efficace et GoogleTest est l'un des plus utilisés et des plus pratiques.

Pour l'affichage graphique des résultats des calculs du moteur de jeu, nous avons choisi d'utiliser FreeGlut, étant une version libre et plus récente de Glut. Cependant, au cours de l'implémentation, il nous est apparu que FreeGlut utilise une version dépassée d'OpenGL, nous allons donc sans doute passer sur GLFW pour les prochaines itérations.

## Difficultés rencontrées

La première difficulté que notre équipe a rencontré est la maîtrise du langage C++. Bien qu'ayant fait quelques projets dans ce langage, s'y replonger après avoir fait du Java ou du python a nécessité quelques mises à jour. Clion nous a aussi posé quelques problèmes notamment l'installation de freeglut avec CMake. Il nous a fallu apprendre la grammaire de Cmake pour qu'il accepte la librairie FreeGlut.

De plus, rajouter GoogleTest nous a causé les mêmes problèmes, d'autant que cette librairie ne possédait pas de .lib et qu'il fallait donc compiler le code pour en obtenir un. Des mises à jour ont dû être faites sur le fichier en .lib pour qu'il accepte la configuration de nos différents PC.

Le point le plus difficile de cette itération fut cependant sans conteste l'utilisation de FreeGlut, OpenGL étant une technologie totalement inconnue pour nous. Réussir à faire déplacer un pixel d'un point à l'autre de l'écran a été plus compliqué que ce que nous pensions notamment dû au fait que la librairie FreeGlut est ancienne et que peu de documentation ont été trouvés sur le net concernant cette librairie.

## Astuce de programmation

Pour plus de clarté, nous avons décidé de mettre un intégrateur dans la classe Particule afin de pouvoir y accéder plus facilement et nous avons mis en place un intégrateur global qui appellerait les intégrateurs de chaque particule.

Pour nous permettre de voir la particule évoluer dans un temps relativement long sans toucher aux paramètres initiaux de vitesse, d'accélération et de damping, nous avons décidé de faire reculer la caméra de FreeGlut pour que celle-ci nous affiche une plus longue distance.